BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

IMAGES ARE BEST AVAILABLE COPY.

As rescanning documents *will not* correct images, please do not report the images to the Image Problems Mailbox.

Annotated reference list agents

Blight, D.C.
TRlabs, Winnipeg, Man.;

This paper appears in: **WESCANEX 97: Communications, Power and Computing. Conference Proceedings., IEEE**
05/22/1997 -05/23/1997, 22-23 May 1997
Location: Winnipeg, Man. , Canada
On page(s): 7-12
22-23 May 1997
Number of Pages: ix+351
INSPEC Accession Number: 5761466

Abstract:
The agents described in the article were used to create a system which does . of the creation, addition, and maintenance of an annotated reference list pag system has been used to maintain a telecommunications Web page for the la years. The strength of this system is that it automatically locates a large num potential links which may be included in the Web page, and uses a fairly fast method of interfacing with the list maintainer to verify links before their inclu The paper discusses the implementation of agents to automate the task of maintaining an annotated reference list

Index Terms:
Internet cooperative systems human factors interactive systems online front-ends software agents user interfaces

7 647

Annotated Reference List Agents

David C Blight , TRLabs, Winnipeg, Canada

Abstract:

The agents described in this paper have been used to create a system which does most of the creation, addition, and maintenance of an annotated reference list page. This system has been used to maintain a telecommunications web page for the last two years. The strength of this system is that is automatically located a large number of potential links which may be included in the web page, and uses a fairly fast method of interfacing with the list maintainer to verify links before their inclusion. This paper discusses the implementation of agents to automate the task of maintaining an annotated reference list.

Keywords: Agents, World Wide Web

1 Introduction

The World Wide Web (WWW) provides an environment in which an unprecedented amount of information is available, however there are no standards or central authority which governs the presentation and format of available data. In fact on the web, the problem is compounded as information providers usually attempt to make a visually appealing page, designed for human interaction, not automated management. While information can be retrieved very efficiently, the classic problem of finding the information still exists. The problems of coping with such a vast amount of information is not new and has been studied in many other environments (libraries, archives, DNA sequencing) however these approaches are based on centralized management and standards which do not apply (and should not be applied) to the internet. This research project involves providing an assistant to locate and manage a subset of the information available on the internet using the available tools and protocols of the internet.

There are traditionally two approaches taken in locating information on the WWW. The first is to use search tools which contact one of more databases and retrieve a list of web sites which match the search criteria. The other approach is to consult a reference page related to a particular subject matter and select appropriate links. Both methods are not totally effective. Search tools are usually limited in their search criteria to pattern matching words and phrases from the available documents with no appreciation for content of the returned links. Reference pages on the other hand are normally person generated lists of references sites and information, but are usually incomplete and out of data due to the enormous effort required to maintain an up to date list.

1.1 Agent Technology

Agents are a software methodology in which software agents are created which interact with systems for a user. On of the key distinctions between agents and general software applications and tools is the interface used in the communications. With software applications, a user interfaces to the applications, which uses computer-computer protocols directly for communication with other applications. Agents on the other hand interface with users, and other agents and applications through the human-computer interfaces. This allows agents to be autonomous systems which interact with existing systems for the benefit of the user.

ISBN 0-7803-4147-3/IEEE CAT. NO. 97CH36117

1997 Conference on Communications, Power and Computing
WESCANEX'97 Proceedings; Winnipeg, MB; May 22-23, 1997; pp. 7-12

1.2 Motivation/Example

This research activity was motivated by the need of the author to find information on the web related to telecommunication and Asynchronous Transfer Mode (ATM) technologies. During the initial period of WWW deployment, not many sites existed on the web, and very little information was available related to these networking technologies. For this reason a reference page was developed to contain pointers to information on telecommunications and ATM technology.

As interest in the web and in telecommunications grew, more and more information became available, and it became impossible to maintain the up to date telecommunications list. Current searches for ATM will return over 100,000 matches. It is not possible manually go through this many links. The large number of new links become available each day, the large number of links which become invalid, and the changing of information in links creates a maintenance nightmare.

In an attempt to automate the maintenance of this page, the following criteria of the automation agents were determined:

- The interface to search tools must be automated. More links must be processed than can be done in reasonable time by the list maintainer. In addition, localized searches must be done on reference pages already found to eliminate the need for needed a large number of similar links.
- Filters must be used to eliminate inappropriate links. ATM commonly refers to Asynchronous Transfer Mode (ATM) and Automated Teller Machines (ATM). The agents must be capable of distinguishing the meaning of ATM.
- The generated list must be annotated. A collection of links is not any more useful than the results of a search engine. Proper annotation of the list makes the web page useful as a research reference page
- Automated maintenance of the page must include period checks of the links to ensure their validity. Many reference pages are only temporary (student created), often moving (corporate pages), or reorganized.

2 Agent Model

With these goals in mind, a system of agents has been implemented to automate the administration and maintenance of the web page. This systems includes agents for:

- interfacing to search tools
- localized search and link collection
- validation of links
- maintenance of web pages.
- manual user interface

3 Implementation

This system is implement as a set of agents, each with specialized tasks. A framework for the interaction of agents and its resources is shown in Figure 1.

3.1 Search Tools Interface Agent

There are three sources of information used in construction of these web pages: search tools which search for specified strings using pattern matching; links available in references pages, and user specified links (usually found from USENET news groups or printed sources). This section discusses how an agent was created which interfaces to search engines to retrieve links matching specified phrases.

In keeping with the goals of this entire system, the system must be simple and use existing tools when possible, the interface to the search engine was implemented using a PERL

different pages.

3.2 Reference Page Agent

Search engines are not the only place to obtain links. Many of the URLs in the WEB page will contains links to reference lists or pages which are being updated fairly regularly. In order to reduce the number of reference pages which have to be maintained in the final reference list page, an agent was created to extract all the links found in other reference pages, and add these to the reference page in a similar manner to those found from search engines.

The only difficulty to this approach is the large number of web URLs in the current reference page may be very large (currently the telecom reference page has about 1000 references). In addition, each reference link may be hierarchical, and sub pages must be found. To accomplish this an agent was written which will check all the sub pages from an URL (those URLs which contain the origonal URL plus further file specifications), and retrieves all external links form that sites (those URLs with different hosts specified). The list of URLs returned is processed similarly to those found from search engines.

3.3 URL Insertion Agent

Once the new links are located, they may be added to the main list. The biggest difficulty is in deciding if the link is appropriate, and in which category it should be added. The URLs in the list collections from the search agents is read by this agent. Since the list is sorted alphabetically by URL, with hostname at the start, URLs from the same host will be located sequentially in this list. When an URL is read from the list, it is first checked to see if the URL is still valid (the reference page may have moved since it was added to the list). The next URLs in the list are also checked to see if they are from the same host. A list of all URLs from the same hosts are presented to the user, who selects one (defaults to first).

Once an URL is selected, it is important for the user to see the page associated with the URL. This agent interfaces with an already running netscape browser through the netscape remote command

netscape -remote 'openURL(

http://www.ee.umanitoba.ca/)'

This causes netscape to show the specified URL. Often the URL found in the lists is not the best one to be placed in the web page. As an example, if searching for ATM, a page may be found which discusses a specific ATM product. It makes more sense to add the base URL (company homepage) rather than this specific piece of information (the lower level pages tend to change more, and are greater in number).

The user is then prompted if he wishes to enter this URL into the main web page. If selects yes, he may change the URL, title, description, and keywords which are extracted from the selected web page. After this information is collected, the agent then inserts the reference into the main and annotated lists in the appropriate spots.

3.4 Web Page Maintenance

Once a web page is created, that is not the end of work that most be done on it. It is very common for many of the links referenced to become invalid, or move. As a result, regular checking of the web page must be performed.

Three agents have been created for maintaining the web pages. The first agent is a very simple agent which removes links. This agent is given an URL, and it checks all relevant web pages for the presence of the URL, and removes the matching links.

A second agent is responsible for checking the integrity of the links. Since the format of these web pages has each link in at

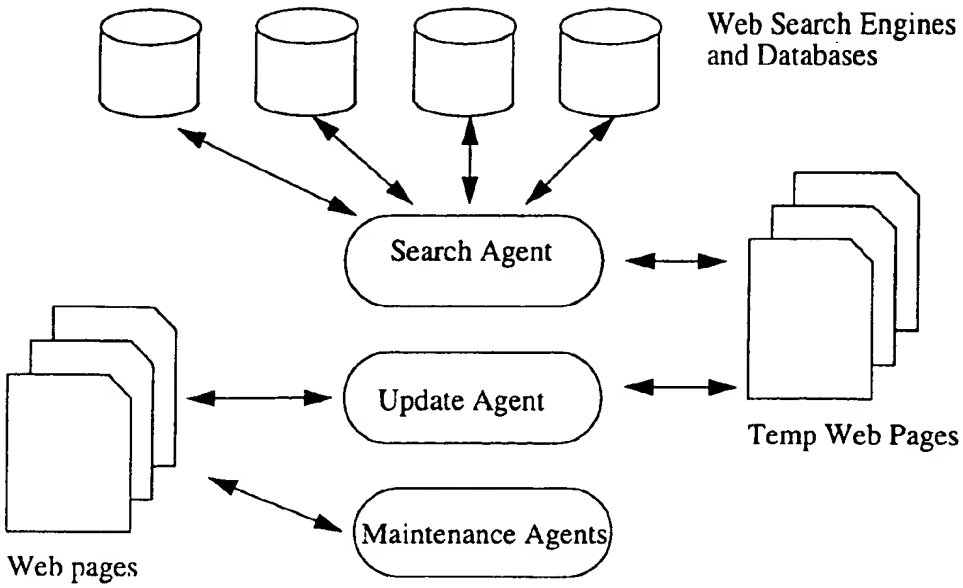

Figure 1: Agents for Annotated List Generation

package WWW::Search which contains a module providing an interface to different search engines[4]. With the aid of this package, it was possible to construct a simple agent written in PERL which contacts the specified search engines, with a list of phrases to search for, a list of rejected sites, and a list of local web pages where the results may be stored.

The search agents have to be careful not to keep locating the same links each time they do a search. To prevent this a list of rejected sites is maintained which contains a list of sites which have been examined, and rejected entry into the reference list. Once a list of links is obtained from the search engine, each link is checked to see if it is in either the reference page, or the reject list. If it has been inserted into the reference page already or has been previously rejected, it will be discarded. If it is a new link, the validity of the link will be verified, and it will be added to the appropriate temporary links page.

Validity of the link is verified by loading the URL. This is accomplished by the PERL package LIBWWW-PERL-5 [5]which contains routines for retrieving http URLs. If the URL is no longer valid (search engines often return pages which have moved or been deleted), and error will be detected, and the URL discarded. If the URL is successfully loaded, the html page returned is parsed for the HTML title. The URL and title are added to a list.

The list of new URLs is next augmented by URL is the found in previous searches which have not been processed. Once a complete list of URLs is obtained, the URLs are sorted by hostname. This allows easy identification of duplicate entries, and will allow easier parsing of similar entries later. If any duplicates are identified, they are removed. The list is next written to a web page. Examples of these pages can be found in [6][7]. It is important to note that lists from multiple search phrases may be stored in a single web page, or separated into

least two pages (the main list, and the categorized annotated list), it is important to check that each page has all the links it is suppose to contain. A agent reads the files, and matches up the links so verify that each entry is in two lists, and in the correct lists.

A third agent exists which verifies the that URLs in the lists are still valid. This is accomplished using the LIBWWW-PERL-5 [5] package. Each URL in the list is retrieved. If no error is detected, the URL remains in the lists. Many types of errors occur in the retrieval of an URL. If the link returns an http error 404 (URL not found), the URL is removed from the list. If the http error 408 (Request Timeout) is received, the date is recorded. This error may result from the web server specified by the URL being down. If this error occurs continually for a week, the URL is removed. The server is also pinged to see if it responding. Other errors are possible, except less frequent. An error 500 (Internal Server Error) can occur due to multiple circumstances, but usually indicated the DNS entry for the server is no longer valid. If a ping is unsuccessful (can not resolve name), the URL is removed. Other errors are left for manual intervention.

3.5 Web Page Syntax Checking

A final step in the creation of a web reference list is the checking of the HTML for syntax errors. Although the simplest means to check a web page is to simply view it with the browser of choice, this is not sufficient as there are many different choices of viewers available, each of which may show the page in a visually distinct way, or many not support some of the HTML features used.

The pages created by these agents are unlikely to introduce many HTML syntax errors as agents generate almost identical HTML each time they are utilized. It is important however to check the HTML, as this helps ensure accessible pages, and can be used to detect errors in processing which might come from the agents.

Two tools are available on the internet which can perform automated checking of html syntax[2][3]. This tools are accessed through URLs, and present the user with a form to enter the URL of the web page to be checked. In a similar manner to how search engines can be queried, these tools can be used to verify the syntax of the HTML page.

4 Results

It is difficult to measure the effectiveness of such a system, as its success is dependent upon subjective measures of the user. This system was developed with maintaining a specific page in mind, although it is extendable to other subjects.

The first issues which should be addresses is the use of agents in this projects. Although the origonal goals was not to make a agents based system, the end design does meet some of the criteria of an agent based system. The distinguishing feature of this system which allows it to be classified as an agent is that its operation is autonomous from the user (except for the actual page updating agent which is more of an assistant than an agent). The agents which do the initial searching and maintenance are capable of acting independently, and interact with the same environment as the user would (the web and search tools).

One of the biggest difficulties in this system has to do with interacting with the list maintainer. The current implementation uses a text based prompting system, but it is currently planned that this will be replaced by a HTML interface (using CGI scripts). The main difficulty in this approach is that it is difficult to get information from the browser being used. There is no facility in web based browsers which allow external agents to determine its current location.

5 Conclusions

The agents described in this paper have been used to create a system which does most of the maintenance of an annotated reference list page. This system has been used to maintain a telecommunications web page for the last two years. The strength of this system is that is automatically located a large number of potential links which may be included in the web page, and uses a fairly fast method of interfacing with the list maintainer to verify links before their inclusion.

The disadvantage of this system is that it still requires manual interaction with the list maintainer. No reliable system for automatic filtering and classification has been found which would allow this step to be eliminated. In addition, the rapid acceleration in web use has created an information overflow problem within the created page. For a reference page on telecommunications, too many links is as much as a problem as dealing with search engines. This is being addressed by eliminating pages which contain only redundant links.

Classification of information is the most challenging of the problems. As the amount of information grew, more categories, and more levels of hierarchy were required. A need for a dynamic classification system which restricts the size of each classification to manageable number of links is required. In addition, each reference link may be classified by multiple categories.

The work here represents a first step in the automated collection and processing of web based references, and has identified a general framework from which to start, and has identified some of the major problems with this approach.

6 References

[1] Telecommunication Information, http://www.ee.umanitoba.ca/~blight/telecom.html

[2] Doctor HTML,http://www2.imagiware.com/RxHTML/

[3] A Kinder, Gentler Validator, http://ugweb.cs.ualberta.ca/~gerald/validate/

[4] AutoSearch WEB Searching, http://www.isi.edu/lsam/autosearch/

[5] LIBWWW-PERL-5, http://www.sn.no/libwww-perl/

[6] New Telecom References, http://www.ee.umanitoba.ca/~blight/telecommunications/telecom-new.html

[7] New ATM References, http://www.ee.umanitoba.ca/~blight/telecommunications/atm-new.html